

Gestione della Complessità



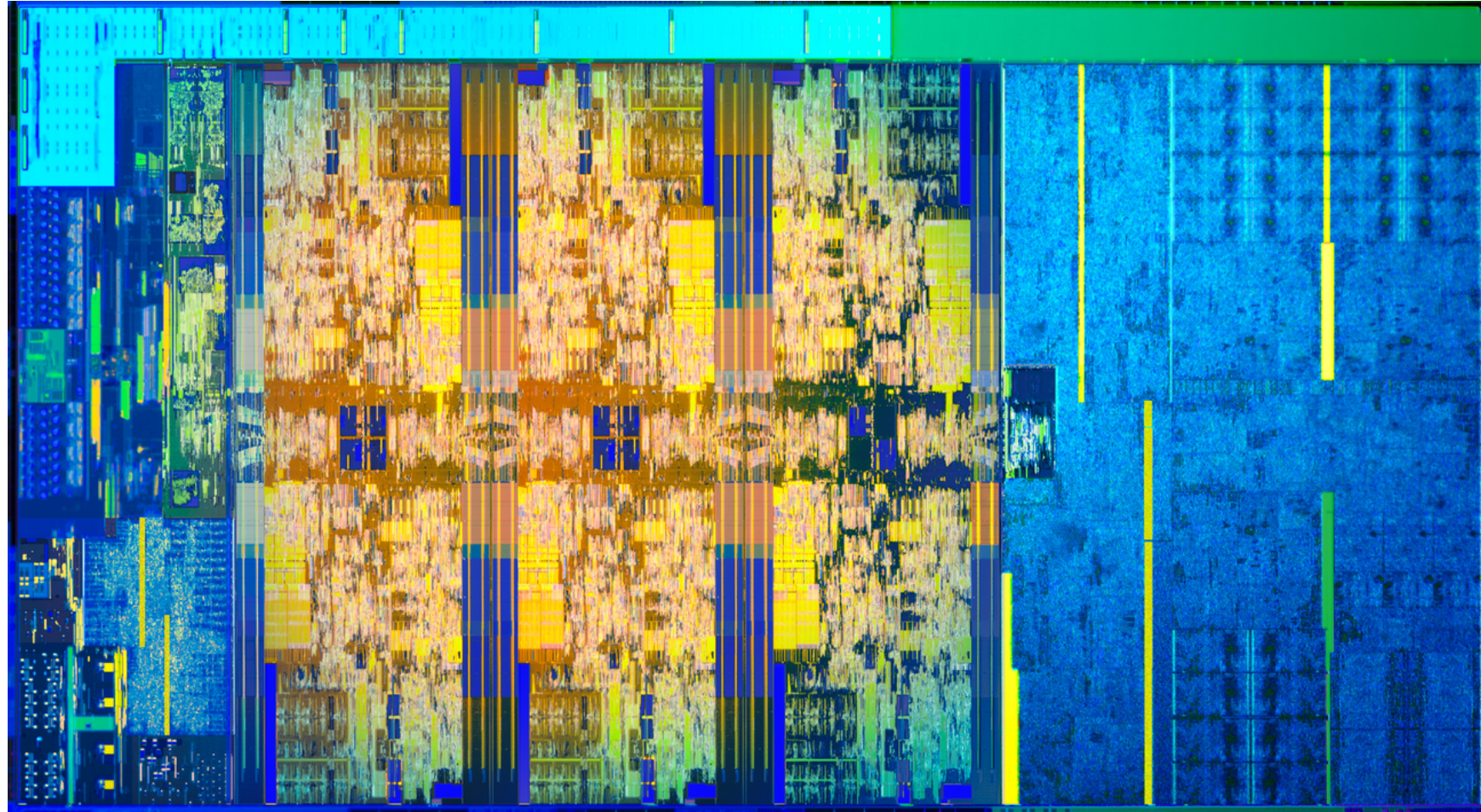
Questioni di Scala



"A single car has about 30,000 parts, counting every part down to the smallest screws"
(Toyota, children's question room)



Questioni di Scala



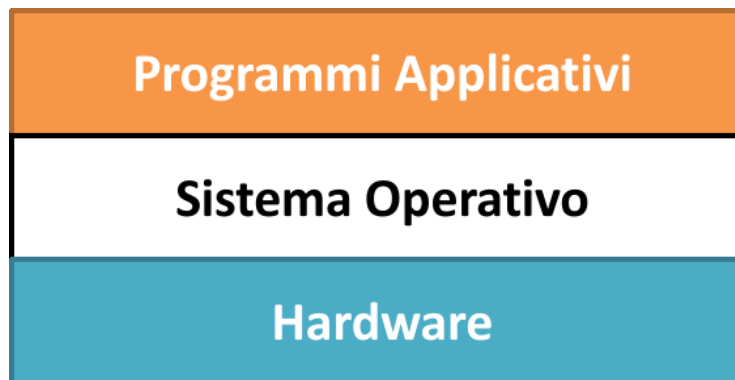
"A single Sky, Kaby or Coffee Lake CPU core and its caches has approximately 217 million transistors"

(anon. utente, dopo un calcolo grossolano)



Livelli di Astrazione

Come gestire tale complessità? Si opera per **livelli di astrazione**

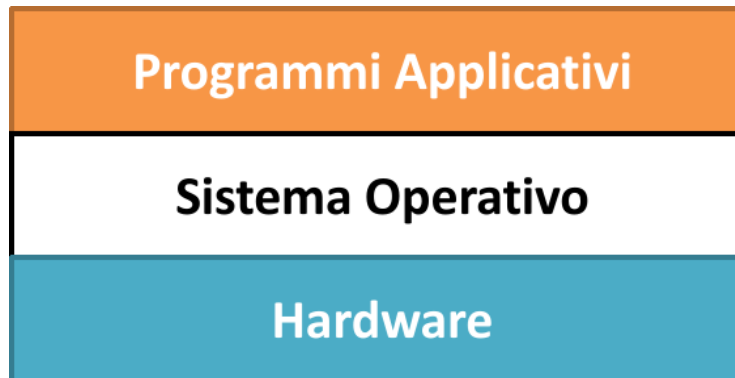


- L'**hardware** è l'elaboratore (e.g. Dell XPS 15' 2019)
- Il **Sistema Operativo** è un insieme di programmi per gestire l'elaboratore
 - E.g. Windows, OSX, Linux/Unix...
- I programmi applicativi sono quelli che usiamo di solito
 - E.g. MS Word, Esplora Risorse, Music Player, IDE...



Livelli di Astrazione

Come gestire tale complessità? Si opera per **livelli di astrazione**

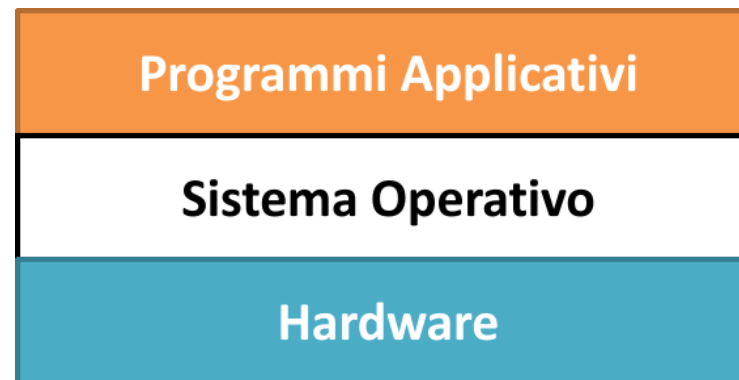


- L'hardware è la vera e propria **macchina fisica**
- Il Sistema Operativo realizza su di esso una **macchina virtuale**
- I programmi applicativi si avvalgono delle funzionalità del SO



Livelli di Astrazione

Come gestire tale complessità? Si opera per **livelli di astrazione**



- Esistono livelli intermedi, ma non li considereremo
 - E.g. firmware (tra HW e SO)
 - E.g. driver di dispositivo (tra firmware e SO)
 - E.g. runtime/frameworks (tra SO e applicativi)
- In questa lezione presenteremo brevemente l'**architettura dell'elaboratore**
- ...E discuteremo alcune delle **funzionalità del Sistema Operativo**



Architettura dell'Elaboratore



Macchine di von Neumann

Un elaboratore elettronico:

- È organizzato in unità funzionali
- L'architettura è ispirata alla "Macchina di von Neumann"
- Si tratta di un modello di elaboratore degli anni '40

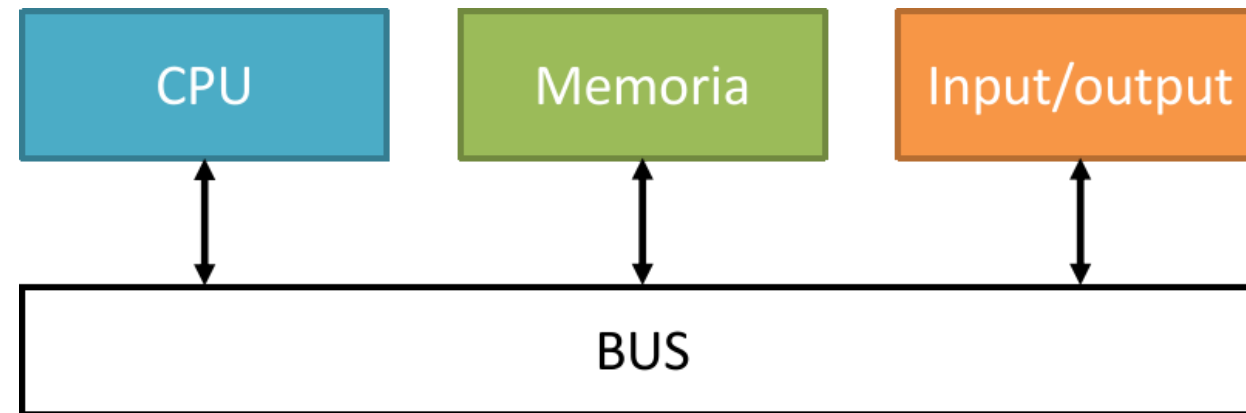


John Von Neumann



Macchine di von Neumann

Una rappresentazione per la Macchina di von Neumann



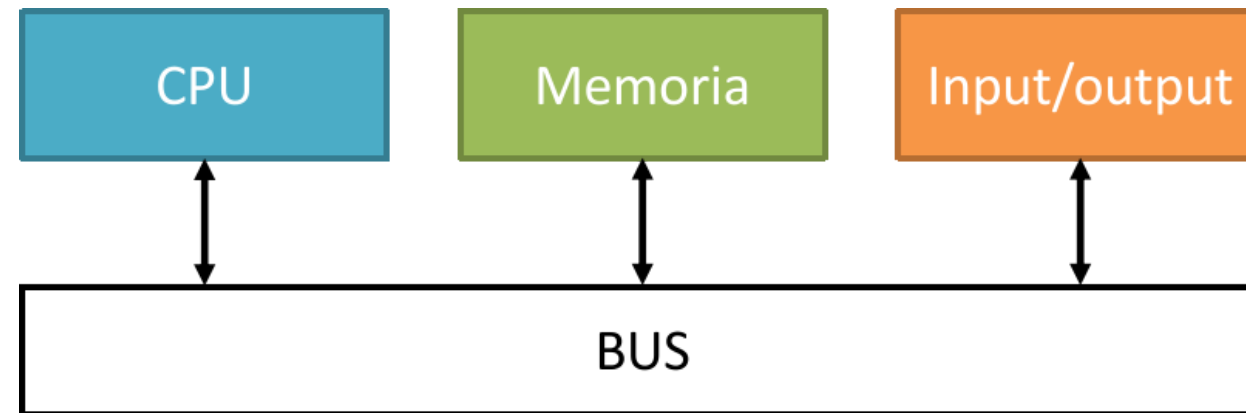
Quattro tipologie di unità funzionale:

- Processore (Central Processing Unit – CPU)
- Memoria Centrale (RAM & ROM)
- Unità di Input/Output (I/O)
- Bus di sistema (per la comunicazione tra componenti)



Macchine di von Neumann

Una rappresentazione per la Macchina di von Neumann



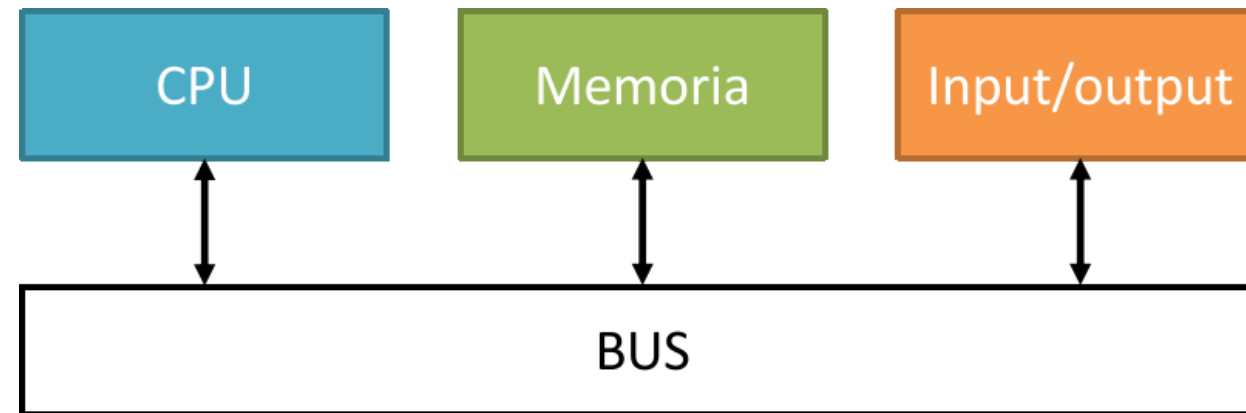
La **CPU** è il cervello del computer

- Può eseguire un insieme di **istruzioni** elementari
 - Velocità determinata dalla **frequenza di clock** (e.g. 4.9GHz)
- Può manipolare un insieme di **tipi di dato primitivi**
 - Questi sono rappresentati mediante sequenze di 0/1 (Bit)
 - 8 Bit equivalgono ad un Byte



Macchine di von Neumann

Una rappresentazione per la Macchina di von Neumann



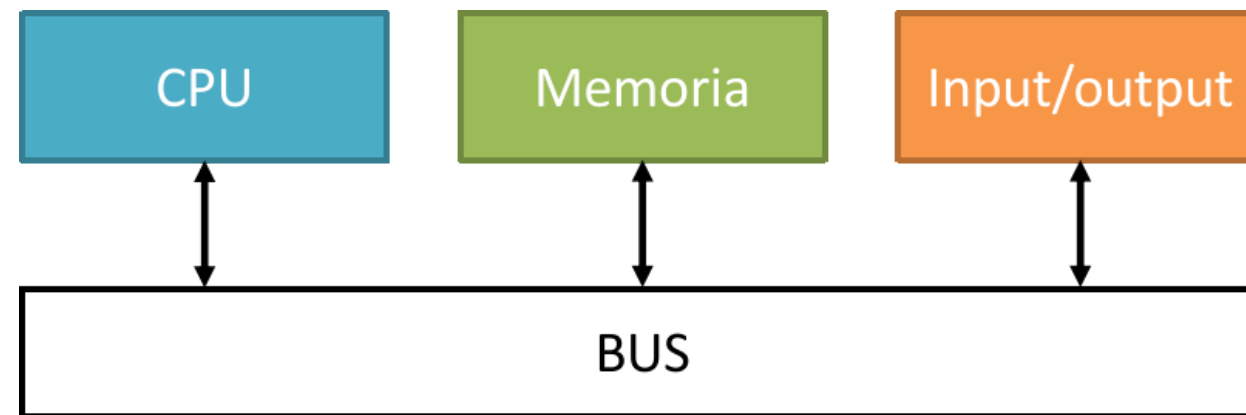
La **CPU** è il cervello del computer

- Può memorizzare una **quantità limitata** di informazioni
 - Pochi MB, nella cosiddetta **memoria cache**
 - ...Ma con accesso molto veloce (e.g. meno di 1 ns)
- **Non** può memorizzare informazioni in modo persistente (è volatile)
 - Quando il PC si spegne il contenuto della cache è perso



Macchine di von Neumann

Una rappresentazione per la Macchina di von Neumann



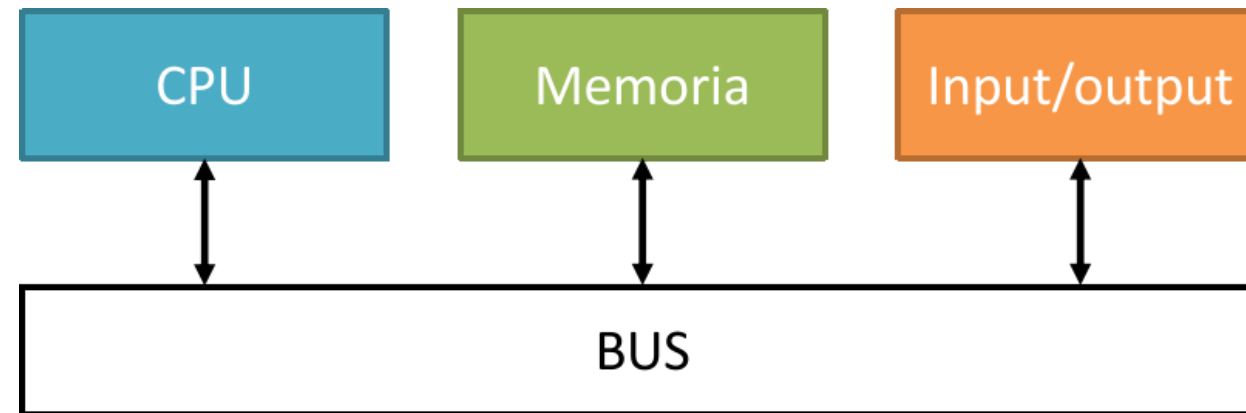
La **memoria centrale** è la "memoria a breve termine" del computer

- Può memorizzare una **quantità considerevole** di informazioni
- È costituita dalla **Random Access Memory (RAM)**...
 - Alcuni GB di capienza, accesso in ~10ns
 - È volatile (il suo contenuto si perde a PC spento)
- ...E dalla **Read Only Memory (ROM)**
 - Alcuni MB, contiene i programmi necessario all'avvio del PC (BIOS, EFI)



Macchine di von Neumann

Una rappresentazione per la Macchina di von Neumann



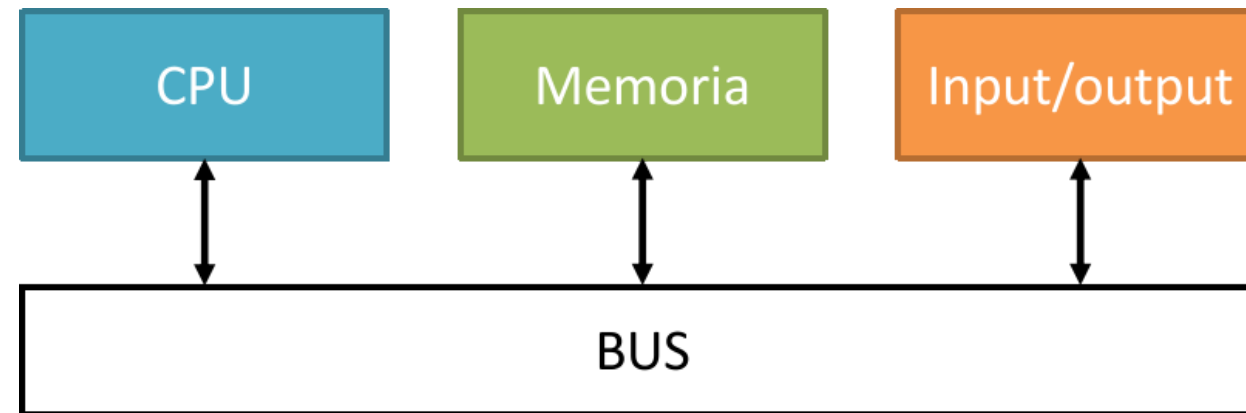
Le **unità di I/O** includono (quasi) tutto il resto

- Dispositivi di input (e.g. tastiera, mouse, superfici sensibili al tatto...)
- Dispositivi di output (e.g. monitor/display, stampante...)
- Scheda di rete, etc.
- ...Ma soprattutto la **memoria di massa** (di cui parleremo tra poco)



Macchine di von Neumann

Una rappresentazione per la Macchina di von Neumann



Il **bus** permette agli altri componenti di comunicare

- È costituito da componenti fisici (chip e connessioni elettriche)
- La comunicazione si avvale di protocolli standard
 - E.g. PCI-express, SATA, USB...



Memoria di Massa

La **memoria di massa** è una serie di dispositivi di I/O

- ...Il cui scopo è memorizzare informazioni **in modo persistente**
- Al prezzo di essere **molto più lenta** della memoria centrale

Può essere realizzata con diverse tecnologie, e.g.

- Dischi magnetici (dischi rigidi, i vecchi floppy disk)
- Memoria flash (Solid State Drive, "chiavette" USB)
- Dispositivi ottici (CD/DVD)

Ogni tecnologia si distingue per:

- Tempo di accesso (i.e. tempo per iniziare una lettura/scrittura)
- Transfer rate (#byte letti/scritti in media per secondo)
- Capacità e costo per byte



Memoria di Massa

Nei dischi magnetici

- I bit sono realizzati mediante lo stato di aree magnetiche (N/S)
- Tali aree sono disposte su un disco rotante (relativamente lento)



- Il tempo di accesso tipico è di 5-10 msec
- La transfer rate tipica è di 100-200 MB/s
- Costo tipico: 15-25\$ per TB
- Capacità tipica: 4-16TB



Memoria di Massa

Nelle memorie flash

- I bit sono realizzati mediante circuiti integrati (transistor)



- Il tempo di accesso tipico è di 25-100 μsec
- La transfer rate tipica è di 500 MB/s in lettura e 200 MB/s in scrittura
 - Le chiavette sono molto più lente!
- Costo tipico: 100\$ per TB
- Capacità tipica: 64GB-4TB



Memoria di Massa

Dischi ottici

- I bit sono realizzati mediante l'orientamento di superfici riflettive



- Il tempo di accesso tipico è di 100-200 msec
- La transfer rate tipica è di 1.2 MB/s
- Costo tipico: 25-35\$ per TB
- Capacità tipica: 4.7-8.5GB



Oltre la Macchine di von Neumann

I calcolatori moderni deviano dalla macchine di Von Neumann

Alcune delle differenze più significative:

- Bus multipli
 - E.g. bus memoria + bus I/O
- Presenza di processore/acceleratori dedicati:
 - Graphical Processing Unit (GPU)
 - Tensor Processing Units (TPU)
- Esecuzioni in pipeline (l'esecuzione è divisa in stadi)
 - Più istruzioni possono essere in esecuzione contemporaneamente
- Architetture parallele
 - E.g. core multipli, sistemi multiprocessore

